

## **Závěrečná zpráva**

**COLLABORATORS**

	<i>TITLE :</i> Závěrečná zpráva		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Martina Vitovská	June 16, 2014	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

## Contents

<b>1</b>	<b>Moje práce na projektu</b>	<b>1</b>
<b>2</b>	<b>Závěr</b>	<b>1</b>

## 1 Moje práce na projektu

Mým úkolem na projektu bylo vytvořit API pro práci s uloženými analýzami ve formátu xml a také funkce pro převod instancí třídy FileInfo do DOM stromu a naopak. Podílela jsem se částečně také na unit testování.

Třída XMLFileManager slouží pro práci s uloženými analýzami. Bylo zapotřebí implementovat metody pro uložení DOM stromu, který představuje jednu analýzu, do souboru xml a vyhledání a smazání již uložených analýz. Třída také umožňuje vyhledat názvy všech doposud vytvořených analýz. Při práci jsem využila hlavně třídu java.io.File a její metody.

Třída FileInfoConverter umožňuje převod instance třídy FileInfo do XML DOM stromu a naopak. Obě tyto metody jsou implementovány pomocí rekurze. DOM strom po vytvoření obsahuje elementy directory a file s příslušnými atributy (například jméno, velikost apod.). Protože se všechny analýzy ukládají jako XML soubory, bylo k nim také potřeba vytvořit XML schéma.

Při testování se objevil problém s vytvořením dat, která by mohla být testována. Tento problém jsme nakonec vyřešili pomocí setUp() a tearDown() metody. V setUp() metodě se před testem vytvoří dočasná adresářová struktura pomocí třídy java.io.File, která je následně po testu pomocí tearDown() metody zrušena. Dalším problémem bylo testování symbolických odkazů. Aplikace má být použitelná na jakémkoliv operačním systému a funkce pro vytvoření symbolického odkazu na systému Windows selhala.

## 2 Závěr

Tento projekt mě kromě práce s nástroji pro xml naučil hlavně týmové práce. Projekt v týmu je něco úplně jiného než samostatný projekt a myslím si, že tato zkušenost byla největším přínosem.

---